H-879
319900189US1M

Title of the Invention

MEMORY CARD

Inventors

Kenji KOZAKAI,

Yuusuke JONO,

Motoki KANAMORI,

Kazunori FURUSAWA,

Atsushi SHIKATA,

Yosuke YUKAWA.

MEMORY CARD

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to memory cards such as file memories, and relates to a technique advantageously applied to, for example, a memory card in which functions of a file memory is included on a single chip.

2. Description of the Related Art

A file memory is a memory card capable of storing file data utilizing a technique similar to management of file allocation by using an FAT (file allocation table) in a hard disk. For example, an electrically rewritable flash memory is used as file data storage area in such a file memory. When file data are accessed, the data are temporarily stored in a buffer memory. For example, file data to be written stored in the buffer memory are written in the flash memory after being assigned with an ECC code at an ECC circuit, and file data read from the flash memory and stored in the buffer memory are output to the outside after error check and correction using an ECC code.

A file memory frequently incorporates a data processor such as a microcomputer for purposes including file management and control of access to the

buffer memory.

PCMCIA-ATA type flash memory cards which are one type of file memories are described on pages 78 and 79 of "Nikkei Electronics" published on April 11, 1994.

SUMMARY OF THE INVENTION

The inventor made a study on a control program area of a file memory having a data processor. A file memory requires a program for debugging or testing in addition to a program for normal file management. The required programs are normally incorporated in a memory card even when the file memory incorporates a data processor such as a microprocessor because such a data processor does not need a function of accessing outside of the memory card. The storage capacity of a ROM for storing programs is thus increased by the program for debugging and testing and the like, which results in a problem in that the scale of the circuit is increased. Especially, the inventor found that a countermeasure muse be taken when limitations on a chip size and the like do not allow a random increase of the storage capacity of a ROM in implementing functions of a memory card such as a file memory in the form of a semiconductor integrated circuit by loading them on a single chip.

It is an object of the invention to provide a

memory card in which a data processor incorporated therein can be caused to execute new programs for purposes including testing or debugging without adding a separate program memory.

The above and other objects and novel features of the invention will become apparent from the description of this specification and the accompanying drawings.

Typical aspects of the invention disclosed in this application can be summarized as follows.

There is provided a memory card 1 comprising an electrically rewritable non-volatile memory 4, a data processor 3 having a function of executing instructions capable of managing the allocation of file data in the non-volatile memory, an interface control circuit 2 having a function of establishing external interface, for controlling the execution of instructions by the data processor in response to external commands and for controlling access to the non-volatile memory and a buffer memory 7 for temporarily storing the file data, in which the buffer memory can be used also as a program memory. Specifically, there is provided command control means 24, 26 for decoding a first command CMD1 supplied from the outside and for instructing the data processor to

fetch an instruction from the buffer memory and to operate. This makes it possible to cause the integrated data processor to execute new programs for purposes including testing or debugging without adding a separate program memory.

Interrupt may be used as a method of control for causing the data processor to execute a program PGM1 stored in the buffer memory. In this case, the command control means may employ a configuration in which an interrupt is requested to the data processor and a first cause of interrupt is notified to the same by decoding the first command.

When vector control is used as a method for controlling the interrupt, the data processor includes a central processing unit 30 capable of responding to an interrupt by transferring the process to an instruction address indicated by a vector retrieved from a vector table 340 according to the cause of interrupt and a ROM 34 to be accessed by the central processing unit. The ROM 34 includes the vector table 340 and a program area 341, and the vector table includes a first vector VCT1 associated with the first cause of interrupt. Thus, the central processing unit can execute an instruction from the beginning of the program in the buffer memory indicated by the first

vector.

The program PGM1 may be transferred to the buffer memory from the outside or from the integrated flash memory. The usability of the file memory is improved by allowing the file memory to transfer the program to the buffer memory by itself. For example, when the program PGM1 is allowed to be stored in the buffer memory from the outside of the file memory, the command control means further requests the data processor an interrupt and notifies it of a second cause of interrupt by decoding a second command CMD2 supplied from the outside. The vector table in the ROM further includes a second vector VCT2 that responds to the second cause of interrupt. The program area of the ROM further includes a transfer control program PGM2 for storing the externally supplied program in the buffer memory starting from a first address thereof. In this case, the second vector is information indicating the leading address of the transfer control program, and the first address is an address that coincides with the address indicated by the first vector VCT1.

When the program PGM1 is allowed to be stored in the buffer memory from the non-volatile memory incorporated in the file memory, the command control

means further requests the data processor an interrupt and notifies the same of a third cause of interrupt by decoding a third command CMD3 supplied from the outside. The vector table in the ROM further includes a third vector VCT3 that responds to the third cause of interrupt. The program area of the ROM further includes a transfer control program PGM3 for storing the program supplied from the non-volatile memory in the buffer memory starting from the first address thereof. In this case, the third vector is information indicating the leading address of the transfer control program, and the first address is an address that coincides with the address indicated by the first vector.

In the memory card 1 constituted by a single chip, even when a random increase of the storage capacity of the ROM is inhibited by limitations on the chip size and the like, the programs for purposes including debugging or testing can be executed within the limitations.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of a memory card LSI which is an example of a memory card according to the invention.

Fig. 2 is a block diagram of an example of a

data processing system that utilizes a card select signal unique to each memory card LSI.

Fig. 3 is a block diagram of an example of a data processing system that utilizes card addresses transmitted along with commands.

Fig. 4 is a block diagram of an example of a host interface circuit.

Fig. 5 illustrates vectors maintained by a ROM and a programmable area of a buffer RAM along with a CPU address map.

Fig. 6 illustrates an example of a state of execution of an extended program.

Fig. 7 illustrates a state of execution of a first transfer control program for storing an external extended program in a buffer RAM.

Fig. 8 illustrates a state of execution of a second transfer control program for storing an extended program from a flash memory in a buffer RAM.

Fig. 9 illustrates a flow of data during a write of file data into a flash memory using a buffer RAM as a data buffer.

Fig. 10 illustrates a flow of data during a read of file data from a flash memory using a buffer RAM as a data buffer.

Fig. 11 illustrates a flow of data during input

and output of work data to and from a CPU using a buffer RAM as a data buffer.

Fig. 12 illustrates a flow of data during input and output of work data between a CPU and a flash memory using a buffer RAM as a data buffer.

Fig. 13 illustrates a principle of information storage in a flash memory.

Fig. 14 is a circuit diagram of a memory cell array utilizing flash memory cell transistors showing a principle of the configuration thereof.

Figs. 15A, 15B and 15C illustrate examples of conditions for voltages for erase and write operations on flash memory cells.

Fig. 16 is a block diagram of an example of flash memory.

Fig. 17 is a circuit diagram of an example of a static memory cell.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

[Summary of Memory Card LSI]

Fig. 1 shows a semiconductor integrated circuit for a memory card according to an embodiment of the invention. The illustrated semiconductor integrated circuit may be regarded as a system-on-chip type LSI (semiconductor integrated circuit) that constitutes a minimum unit of a file memory and is formed on a

single semiconductor substrate (chip) such as single crystal silicon, although this is not limiting the invention.

A semiconductor integrated circuit (also simply referred to as "memory card LSI") 1 shown in Fig. 1 has an interface control circuit 2, a microcomputer 3 which is an example of a data processor, a flash memory 4 which is an example of an electrically rewritable non-volatile memory, a reset circuit 5, a clock oscillation circuit 6 utilizing an oscillator, a buffer RAM 7 and a work RAM 8.

A power supply voltage Vcc and a ground voltage Vss are externally input to the memory card LSI 1 as operating power supplies. The input power supply voltage Vcc and ground voltage Vss are supplied to each of the above-described circuits.

The interface control circuit 2 has a host interface circuit (Host I/F) 11, a microcomputer interface (Micro I/F) 12, a file control logic (FCL) 13 and a data transfer logic (DTL) 14 which are connected each other by a bus 10.

A clock signal (Clock) 2A and a card select signal (Card Select) 2D are externally input to the host interface circuit 11, and a command (Command) 2B and data (Data) 2C are input to and output from the

same. Each of the command 2B and data 2C is input and output on a bit serial basis, although this is not limiting the invention. The host interface circuit 11 accepts the externally supplied command 2B and decodes it to instruct operations of the microcomputer 3 and flash memory 4 and to control file data access to the flash memory 4.

The instruction of an operation of the microcomputer 3 is performed by supplying an interrupt signal NMI and a cause of interrupt from the host interface circuit 11 to the microcomputer 3 through the microcomputer interface 12. The microcomputer interface 12 exchanges the interrupt signal NMI, a control signal Ctl and various data such as data information and control information with the microcomputer 3.

The file control logic 13 controls file data access to the flash memory 4 according to the result of command decoding at the host interface circuit 11 under control of the microcomputer 3.

The buffer RAM 7 is used as a file data buffer memory for temporarily storing file data externally supplied to the host interface circuit 11 or temporarily storing file data read from the flash memory 4. It is also used as an extended program

memory for the microcomputer 3.

The control over access to the buffer RAM 7 is carried out through the data transfer logic 14. The data transfer logic 14 has an ECC circuit 14A to check and correct errors during access to the buffer RAM 7 with the ECC circuit.

When the buffer RAM 7 is used as a file data buffer memory, in a file data writing operation, the file data are read by the data transfer logic 14 from the buffer RAM 7 on to the bus 10, and the read file data are written in the flash memory 4 under control of the file control logic 13. In a file data reading operation, the file data are read from the flash memory 4 on to the bus 10 under control of the file control logic 13, and the read file data are written in the buffer RAM 7 under control of the file transfer logic 14. The state in which the buffer RAM 7 is used as a buffer memory for file data is achieved when a file access command is externally supplied to the interface control circuit 2; an interrupt according to the result of decoding of the command is accepted by the microcomputer; and the result of the decoding of the command is supplied to the file control logic 13 and file transfer logic 14.

The buffer RAM 7 is mapped to an address space

of the microcomputer 3 (more particularly, a CPU 30 to be described later). The microcomputer 3 can access the buffer RAM 7 similarly to access to the work RAM 8 through the data transfer logic 14. For example, this mode of access is enabled when the buffer RAM 7 is used as an extended program memory of the microcomputer 3. The state in which the microcomputer 3 utilizes the buffer RAM 7 as an extended program memory is realized when an external command for the execution of an extended program is supplied to the interface control circuit 2 and an interrupt according to the result of decoding of the command is accepted by the microcomputer 3. This process will be detailed later.

The file memory LSI 1 has a file data access system which is compatible with a hard disk apparatus, although this is not limiting the invention. For example, one cluster which is a unit area for access management includes four sectors, and a management area is allocated to each of the clusters. A management area has pointer information for determining the arrangement of clusters forming a file, information on the number of rewrites, information identifying good and defective sectors and the like. Further, the flash memory 4 has a directory

area that identifies the file name of a file stored therein and the leading cluster of the same.

In order to manage the arrangement of file data in clusters in the flash memory 4, the microcomputer 3 generates a management table in the integrated SRAM 35 based on information of the management areas and directory area. The microcomputer 3 controls the generation and update of the management table and generates information specifying a management unit area to be accessed at access to file data using the management table. The information for controlling access to file data is supplied to the file control logic 13 through the microcomputer interface 12.

The microcomputer 3 has a central processing unit (CPU) 30, an integrated ROM (read only memory) 34 in which an operation program of the CPU 30 and the like are stored, an integrated SRAM (static random access memory) 35 used as a work area of the CPU 30 or an area for temporarily storing data, a bus controller (BSC) 33 for controlling the bus cycle of an external bus 37 when the CPU 30 accesses an external address space and a user break controller (UBC) 31 for supporting debugging such as breakpoint control, which are each connected to an internal bus 38. An interrupt signal NMI and a cause of interrupt are

input to an interrupt control circuit (INTC) 32 which in turn requests the CPU 30 an interrupt by performing interrupt priority control. An interrupt processing program is stored in the integrated ROM 34, although this is not limiting the invention.

A watch dog timer (WDT) 36 for monitoring the run away of the CPU 30 and the like is connected to the external bus 37 of the microcomputer 3 in addition to the bus controller 33. Further, the work RAM 8 and microcomputer interface 12 are connected through the bus. The microcomputer 3 has one I/O port 39A as another interface circuit. The I/O port 39A is exclusively used for inputting the interrupt signal NMI and outputting control signals represented by the control signal Ctl. No general purpose I/O is provided, although this is not limiting the invention.

The microcomputer 3 has a sleep mode and a standby mode as low power consumption modes, although this is not limiting the invention. The CPU 30 executes a sleep instruction when a standby control bit provided in a control register (not shown) has a first logical value to enter the sleep mode. In the sleep mode, the CPU 30 stops operating with the state of the register and the like kept unchanged. Peripheral circuits continue operating. The sleep

mode is cancelled by an interrupt or reset. The CPU 30 executes a sleep instruction when the standby control bit provided in the control register has a second logical value to enter the standby mode. In the standby mode, the CPU 30 stops operating with the state of the register and the like kept unchanged, and the peripheral circuits also stop operating. The standby mode is canceled by an interrupt or reset.

A clock signal CLK2 is supplied from the oscillation circuit 6 to a clock pulse generator 39B of the microcomputer 3. For example, when the microcomputer 3 is set in the standby mode, the oscillation circuit 6 stops outputting the clock signal CLK2 according to a signal output by the microcomputer 3 in response. When the interrupt signal NMI is asserted from the microcomputer interface 12 to the port 39A in this state, the clock control circuit 15 detects the same state. Accordingly, the clock control circuit 15 causes the oscillation circuit 6 to resume the supply of the clock signal CLK2. Therefore, when the CPU 30 responds to the interrupt, the microcomputer 3 can leave the standby mode because supply of the clock signal CLK2 has already been resumed by then.

The reset circuit 5 resets the interface control

circuit 2 with a reset signal RES1 and resets the microcomputer 3 with a reset signal RES2. The flash memory 4 is reset by a reset signal RES3 which is controlled according to the value of a reset enable bit RSB provided in a control register in the file control logic (FCL) 13.

Figs. 2 and 3 show examples of a data processing system utilizing memory card LSIs 1 as described above. Although not shown, the memory card LSIs 1 are packaged using a technique such as resin molding with connectors thereof exposed. 100 represents a host system, and 101 represents slots for mounting the memory cards. Figs. 2 and 3 illustrate configurations to allow a plurality of memory card LSIs 1 to be mounted at a time. The clock signal line 2A, command signal line 2B and data signal line 2C are shared by the memory card LSIs 1 in both of the configurations. The selection of the plurality of memory card LSIs 1 mounted is carried out using the card select signal 2D specific to each of the memory card LSIs 1 in the example in Fig. 2 and using card addresses transmitted along with commands in the example in Fig. 3. In the example in Fig. 3, a memory card LSI 1 recognizes that it has been selected upon the input of a card address allocated to it at an initializing operation.

Fig. 4 is a block diagram of the host interface circuit 11. Referring to Fig. 4, the host interface circuit 11 has a command input register 20 to which the command 2B is input, a response control circuit 21 for responding to the input of commands, a data input register 22 to which data 2C are input and a data output register 23 for outputting the data 2C. An input command is decoded by a command decoder 24 and, in accordance with the result of the decoding, a control logic circuit 26 controls interrupts to the microcomputer 3, data input and output, responses to a host apparatus and the like. 27 represents a memory for temporary storage used by the control logic 26.

[Execution of Extended Program]

Next, said buffer RAM7 will be described usable constraction as said extended program memory in detail.

For example, an extended program is executed using vector type interrupt control performed by the microcomputer 3. The vector interrupt of the microcomputer 3 is performed as follows. Specifically, the microprocessor 3 is notified of an interrupt by an interrupt signal NMI from the interface control circuit 2. The interrupt control circuit 32 performs interrupt priority control and the

like on the interrupt initiated by the interrupt signal NMI and asserts an interrupt request signal INT to the CPU 30 when the interrupt is accepted. When the interface control circuit 2 detects the acceptance of the interrupt, it supplies information specifying a cause of the interrupt to the external bus 37 through the microcomputer interface 12. The CPU 30 retrieves a vector associated with the cause of interrupt from the vector table. The CPU 30 proceeds to an instruction address indicated by the retrieved vector and branches to a process of responding to the interrupt. In the case of an interrupt for which the process is to return to the state immediately preceding the interrupt after the process of responding to the interrupt, the state is obviously saved before the process of responding to the interrupt.

Fig. 5 shows mapping of the addresses of the integrated ROM 34, work RAM 8, buffer RAM 7 and integrated SRAM 35 to an address space which can be managed by the CPU 30.

A part of the buffer RAM 7 is an area 70 which can be used also as an extended program memory (programmable area), although this is not limiting the invention. A program stored in the programmable area

70 is referred to as "extended program PGM1".

The integrated ROM 34 has the vector table 340 and program area 341. The vector table 340 typically has a first vector VCT1, a second vector VCT2 and a third vector VCT3. The program area 341 has a first transfer control program PGM2 and a second transfer program PGM3 as subroutines. Programs for a reset process, file managing process and the like are also stored, although not shown.

The vector VCT1 has information on the leading address of the programmable area 70. The extended program PGM1 is stored starting from the leading address of programmable area 70. The vector VCT2 has information on the leading address of the storage area of the first transfer control program PGM2. The vector VCT3 has information on the leading address of the storage area of the second transfer control program PGM3.

The first transfer control program PGM2 is a transfer control program for storing an extended program PGM1 externally supplied to the memory card LSI 1 in the programmable area 70 starting from the leading address of the same. The second transfer control program PGM3 is a transfer control program for reading an extended program PGM1 which has been

transferred to the flash memory 4 in the form of a file or which has been stored therein in advance at a manufacturing step and for storing the same in the programmable area 70 starting from the leading address thereof.

Fig. 6 schematically shows a process of executing an extended program PGM1 stored in the programmable area 70. The execution of the extended program PGM1 stored in the programmable area 70 is instructed by an extended program execution command CMD1 externally supplied to the interface control circuit 2. When the extended program execution command CMD1 is input to the command input register 24 of the interface control circuit 2, the command decoder 24 decodes the same and, upon receipt of the result of the decoding, the control logic circuit 26 outputs an interrupt signal NMI and notifies the CPU 30 of a first cause associated with the extended program execution command. After performing a required state saving process and the like, the CPU 30 retrieves a first vector VCT1 associated with the first cause from the vector table 340 and proceeds to the execution of the extended program PGM1 in the programmable area 70.

Fig. 7 schematically shows a process of

executing the first transfer control program PGM2. The execution of the first transfer control program PGM2 is instructed by an external transfer control execution command CMD2 for an extended program externally supplied to the interface control circuit 2. When the external transfer control execution command CMD2 for an extended program is input to the command input register 20 of the interface control circuit 2, the command decoder 24 decodes the same and, upon receipt of the result of the decoding, the control logic circuit 26 outputs an interrupt signal NMI and notifies the CPU 30 of a second cause associated with the external transfer control execution command. After performing a required state saving process and the like, the CPU 30 retrieves a second vector VCT2 associated with the second cause from the vector table 340 and proceeds to the execution of the first transfer control program PGM2.

Fig. 8 schematically shows a process of executing the second transfer control program PGM3. The execution of the second transfer control program PGM3 is instructed by an internal transfer control execution command CMD3 for an extended program externally supplied to the interface control circuit 2. When the internal transfer control execution

command CMD3 for an extended program is input to the command input register 24 of the interface control circuit 2, the command decoder 24 decodes the same and, upon receipt of the result of the decoding, the control logic circuit 26 outputs an interrupt signal NMI and notifies the CPU 30 of a third cause associated with the internal transfer control execution command. After performing a required state saving process and the like, the CPU 30 retrieves a third vector VCT3 associated with the third cause from the vector table 340 and proceeds to the execution of the second transfer control program PGM3.

As described above, it is possible to cause the CPU 30 to execute separate programs for purposes including testing or debugging using the buffer RAM 7 without any additional program memory. In the memory card LSI 1 constituted by a single chip, even when a random increase of the storage capacity of the ROM 34 is inhibited by limitations on the chip size and the like, the programs for purposes including debugging or testing can be executed within the limitations. Further, referring to control over the transfer of an extended program PGM1 to the buffer RAM 7, since the transfer of an extended program PGM1 externally supplied or stored in the integrated flash memory 4

can be controlled by the file memory 1 itself, the file memory 1 has preferable usability with respect to an extended program.

As described above, other modes of data transfer using the buffer RAM 7 include modes of utilization inherent in a file memory in which it is used as a data buffer when file data are written in the flash memory 4 (Fig. 9) and in which it is used as a data buffer when file data held in the flash memory 4 are read out (Fig. 10). There are other modes of utilization of the buffer RAM 7 in which it is used as a data buffer when work data are input and output to and from the CPU 30 as shown in Fig. 11 and in which it is used as a data buffer when work data are exchanged between the CPU 30 and the flash memory 4 as shown in Fig. 12.

[Memory]

An example of the flash memory 4 will now be described for reference. First, a description will be made with reference to Figs. 13A and 13B on a principle of the storage of information in the flash memory.

The memory cell shown in Fig. 13A as an example is constituted by an insulated gate type field effect transistor having a double layer gate structure. In

Fig. 13A, 431 represents a p-type silicon substrate; 432 represents a p-type semiconductor region formed on said silicon substrate 431; and 433 and 434 represent n-type semiconductor regions. 435 represents a floating gate formed above the p-type silicon substrate 431 with a thin oxide film 436 (having a thickness of, for example, 10 nm) as a tunnel insulation film interposed therebetween, and 437 represents a control gate formed above the floating gate 435 with an oxide film 438 interposed therebetween. The source is constituted by the region 434, and the drain is constituted by the regions 433 and 432. Information stored in this memory cell is held in the transistor substantially as a change in a threshold voltage. In the following description, a transistor of a memory cell for storing information (hereinafter also referred to as "memory cell transistor") is of the n-channel type unless otherwise specified.

For example, an operation of writing information in a memory cell is carried out by applying a high voltage to the control gate 437 and the drain and by injecting electrons into the floating gate 435 from the drain side using avalanche injection. As a result of the write operation, as shown in Fig. 13B, the

threshold voltage of the storage transistor as viewed from the control gate 437 becomes higher than that of a storage transistor in an erase state which has not been subjected to a write operation.

For example, an erase operation is carried out by applying a high voltage to the source and by extracting electrons from the floating gate 435 toward the source using the tunnel phenomenon. Shown in Fig. 13B, the threshold voltage of the storage transistor as viewed from the control gate 437 becomes lower by erase operation. As shown in Fig. 13B, the threshold voltage of a memory cell transistor is a positive voltage level in both of the write and erase states. Specifically, the threshold voltage in the write state is higher than a word line selection level supplied from a word line to the control gate 437, and the threshold voltage in the erase state is lower than the same. Since such a relationship exists between the two threshold voltages and the word line selection level, the memory cell can be constituted by a single transistor without using a selection transistor. Since stored information is electrically erased by extracting electrons accumulated in the floating gate 435 toward the source electrode, a continuous erase operation for a relatively long time will extract

electrons in a quantity larger than that of the electrons injected into the floating gate 435 at the write operation. Therefore, when an over-erase is performed in which an electrical erase is continued for a relatively long time, the threshold voltage of the memory cell transistor becomes, for example, a negative level, which results in a problem in that selection occurs in spite of the fact that the word line is at an unselect level. Writing may be carried out utilizing a tunnel current similarly to erasing.

During a read operation, in order to prevent a weak write in the memory cell or unwanted injection of the carrier into the floating gate 435, the voltage applied to the drain and the control gate 437 is limited to a relatively small value. For example, a low voltage on the order of 1 V is applied to the drain, and a low voltage on the order of 5 V is applied to the control gate 437. The magnitude of the channel current flowing through the memory cell transistor is detected by applying those voltages to allow the information stored in the memory cell to be determined as "0" or "1".

Fig. 14 shows a principle of the configuration of a memory cell array utilizing memory cell transistors as described above. Fig. 14 shows four

typical memory cell transistors Q1 through Q4. In the memory cells arranged in X- and Y-directions in the form of a matrix, the control gates of the memory cell transistors Q1 and Q2 (Q3 and Q4) arranged on the same row (selection gates of the memory cells) are connected to a respective word lines WL1 (WL2), and the drain regions of the storage transistors Q1 and Q3 (Q2 and Q4) arranged on the same column (input/output nodes of the memory cells) are connected to a respective data line DL1 (DL2). The source regions of the storage transistors Q1 and Q3 (Q2 and Q4) are coupled to a source line SL1 (SL2).

Figs. 15A, 15B and 15C show examples of conditions for voltages for the erase and write operations on the memory cells. In those figures, the memory elements are the memory cell transistors, and the gates are the control gates as the selection gates of the memory cell transistors. In those figures, erasure based on a negative voltage method is carried out by applying a negative voltage, e.g., -10 V to the control gate to generate a high electrical field required for erasure. As apparent from voltage conditions shown in the figures, erasure based on a positive voltage method allows at least memory cells whose sources are commonly connected to be erased at a

time. Therefore, when the source lines SL1 and SL2 are connected in the configuration in Fig. 14, the four memory cells Q1 through Q4 can be erased at a time. According to a source line division method, data lines may serve as units (common source lines extend in the direction of data lines) as typically illustrated in Fig. 14 or word lines may alternatively serve as units (common source lines extend in the direction of source lines). Erasure based on the negative voltage method allows memory cells whose control gates are commonly connected to be erased at a time.

Fig. 16 shows an example of the flash memory 4. In Fig. 16, 403 represents a memory array which has memory mats and sense latch circuits. The memory mat has a multiplicity of non-volatile memory cell transistors which can be electrically erased and written. For example, the memory cell transistors have a configuration including a source and a drain formed on a semiconductor substrate or in a memory well, a floating gate formed in a channel region with a tunnel oxide film interposed and a control gate overlaid on the floating gate with a layer insulation film interposed as described with reference to Fig. 13. The control gates are connected to word lines

406; the drains are connected to bit lines 405; and the sources are connected to source lines which are not shown.

External input/output terminals I/O0 through I/O7 are also used as address input terminals, data input terminals, data output terminals and command input terminals. X-address signals input through the external input/output terminals I/O0 through I/O7 are supplied to an X-address buffer 408 through a multiplexer 407. An X-address decoder 409 decodes internal complementary address signals output by the X-address buffer 408 to drive the word lines.

Although not shown, the memory mats included in the memory array 403 are configured on the left and right of the sense latch circuit array. Specifically, precharge circuits, bit lines and the like are provided at both of the input and output nodes of the sense latch circuits. The bit lines 405 are selected based on a selection signal output by a Y-address decoder 411 by Y gate array circuit 413. Y-address signals input through the external input/output terminals I/O0 through I/O7 are preset in a Y-address counter 412, and address signals which are sequential increments starting with the preset values are supplied to the Y-address decoder 411.

A bit line selected by a Y gate array circuit 413 is conducted to an input terminal of an output buffer 415 during a data output operation and is conducted to an output terminal of an input buffer 417 through a data control circuit 416 during a data input operation. The connection between the output buffer 415, input buffer 417 and input/output terminals I/O0 through I/O7 is controlled by the multiplexer 407. Commands supplied through the input/output terminals I/O0 through I/O7 are supplied to a mode control circuit 418 through the multiplexer 407 and input buffer 417. The data control circuit 416 is capable of supplying the memory array 403 with not only data supplied through the input/output terminals I/O0 through I/O7 but also data having logical values in accordance with the control of the mode control circuit 418.

A control signal buffer circuit 419 is supplied with a chip enable signal CEb, an output enable signal OEb, a write enable signal WEb, a serial clock signal SC, a reset signal RESb and a command enable signal CDEb as access control signals.

The mode control circuit 418 controls a function of interfacing external signals according to the states of those signals and controls internal

operations according to command codes. When a command or data is input to the input/output terminals I/O0 through I/O7, the signal CDEb is asserted; the signal WEb is asserted further if it is a command; and the signal WEb is negated if it is data. When an address is input, the signal CDEb is negated and the signal WEb is asserted. This allows the mode control circuit 418 to discriminate between commands, data and addresses input through the external input/output terminals I/O0 through I/O7 on a multiplex basis. During an erase or write operation, the mode control circuit 418 can externally indicate such a state by asserting a ready/busy signal R/Bb.

An internal power supply circuit 420 generates various operating power supplies 421 for purposes such as write, erase verify and read and supplies them to the X-address decoder 409 and memory cell array 403.

The mode control circuit 418 controls the flash memory 4 as a whole according to commands. The operation of the flash memory 4 is basically determined by the commands.

The commands allocated to the flash memory include, for example, read, erase and write commands. The read command is constituted by a first command, and the other commands are constituted by a first

command and a second command.

The flash memory 4 has a status register 423 for indicating the internal status thereof, and the contents of the same can be read through the input/output terminals I/O0 through I/O7 when the signal OEb is asserted.

When a write operation is instructed by the write command, the sense latch circuits can latch write data supplied through the Y gate array circuit 413. In this example, since the flash memory 4 has the input/output terminals I/O0 through I/O7 for eight bits, write data can be set in eight sense latch circuits at one cycle of input of write data. In the context of this description, since writing is performed on a word line basis, a write voltage is applied to cause a write operation after write data are set in sense latch circuits associated with the bit lines of all memory cells whose selection terminals are coupled to one word line. At a write operation, for example, all bit lines are precharged to a predetermined level in advance; the bit lines of memory cells selected for writing are discharged down to a ground potential; and the bit lines of memory cells unselected for writing are maintained at the precharge level. When a high write voltage is applied

to word lines selected for writing, a high voltage is applied between the control gates and drains of the memory cells selected for writing to increase the threshold voltage of the memory cells selected for writing, which realizes a write state. Prior to a write operation, the memory cells are in an erase state in which the threshold voltage is low. The threshold voltages for write and erase may be defined reversely.

The reset signal RESb in Fig. 16 is a signal that corresponds to the reset signal RES3 in Fig. 1. The multiplexer 407 and control signal buffer circuit 419 in Fig. 16 exchange input/output signals with the FCL 13 in Fig. 1.

A description will now be made on an example of static memory cells that constitute the integrated SRAM 35, work RAM 8 and buffer RAM 7. Fig. 17 one typical static memory cell 70. The static memory cell 70 has a pair of CMOS inverters formed by an n-channel type MOS transistor 71 and a p-channel type MOS transistor 72, and an input terminal of one of the CMOS inverters is cross-coupled to an output terminal of the other CMOS inverter to form a static latch. A pair of storage nodes of the static latch are coupled to complementary bit lines 78t and 78b through n-

channel type selections MOS transistors 75 and 76. The gates of the selection MOS transistors 75 and 76 are coupled to a word line 77.

While the invention conceived by the inventor has been specifically described based on preferred embodiments thereof, the invention is not limited to the embodiments and may obviously modified in various ways without departing from the principle of the invention.

For example, the program stored in the buffer memory is not limited to a program for testing or debugging and may be a file data compression program or the like. The term "memory card" in the context of the present specification is not meant to exclude other functions, and it is used on an assumption that a memory card at least has a function of storing file data and may include communication interface functions such as those of MODEMs and TAs (terminal adapters), networking functions such as that of LANs (local area networks), video capture functions, voice recognizing functions and the like. Therefore, programs used for such functions may be stored in the buffer memory.

The programmable area is not limited to a partial storage area of the buffer memory, and it may be the entire area of the same.

The above-described commands and data are not limited to serial signals and may be parallel signals.

The cluster size is not limited to four sectors and may be appropriately determined in accordance with the configuration of the memory mats of the flash memory, the storage capacity of the integrated SRAM that develops the management table and the like.

The term "microcomputer" implies logic circuit units having a function of fetching and executing instructions and does not limit the invention to configurations in which a single microcomputer uses verified design data of an LSI associated therewith. The microcomputer may be a circuit having a new customized design.

The memory card LSI has been described as a single chip. The single chip configuration is expected to provide a higher operating speed and lower power consumption in comparison to multi-chip configurations.

Effects that can be achieved by typical aspects of the invention disclosed in this application can be summarized as follows.

Since a buffer memory used for writing and reading file data can be also used as a program memory, separate programs for purposes such as testing

or debugging can be executed using the buffer memory without any additional program memory. In a memory card constituted by a single chip, even when a random increase of the storage capacity of the ROM is inhibited by limitations on the chip size and the like, programs for purposes including debugging or testing can be executed within the limitations. Further, referring to control over the transfer of an extended program to the buffer memory, since the transfer of an extended program externally supplied or stored in an integrated flash memory can be controlled by the file memory itself, the file memory has preferable usability with respect to an extended program.